2nd Annual GSFC-JPL Quality Mission Software Workshop

Goddard Space Flight Center
Jet Propulsion Laboratory

Session 4: Relationships to Projects

San Diego, California
May 16-18, 2000

AGENDA
Session 4: Relationships to Projects

Day 2
Wednesday – May 17, 2000

Relationships to Projects

1:00 pm	Overview of Classes for Software for Project Managers	J. Steinbacher
1:30 pm	Software Quality Assurance	B. Sigal
2:00 pm	Software Development Principles	M. Lavin
2:30 pm	Break	
2:45 pm	Approaches and Technologies for Flight Software V&V	M. Bartholomew
3:45 pm	Summary of Metrics Session	M. Stark
4:00 pm	Working Session	All
5:00 pm	End of Second Day	

2nd Annual GSFC-JPL Quality Mission Software Workshop

Relationships to Projects: Overview of the Understanding Software for Project Management Course

JPL

M.J. Steinbacher

17 May 2000

Training Objective

- Provide Project Management with information that will increase their level of understanding of the software issues related to project management, and thus more effectively address the software challenges in their projects

Course Development

- **Input to the Course**
  - Software Managers, Project Management representatives, and members of Center for Space Mission Information and Software Systems (CSMISS) were interviewed to provide topics of interest for JPL Project Management
- **Course Design**
  - Members of the Mission Software Process (MSP) (part of CSMISS) participated in the design activities
  - A design review was held; the topics presented here represent the course design as of that review
  - Course is focused on software issues

Course Outline

Day 1

- Software and Its Unique Aspects
- Software Life-Cycles and Milestones
- System Engineering Considerations
- Software Technologies
- COTS and Reuse
- Software Acquisition
- Software Evaluation and Quality

Day 2

- JPL, NASA, and Industry - Resources and Standards
- Configuration Management and Documentation
- Staffing Considerations and Workforce Planning
- Understanding Software Planning and Metrics
- Using the Mission Data System (MDS)
- Lessons Learned - Panel

Course Topics

- **Software and its Unique Aspects**
  - Define what software is and how much is developed and used at JPL
  - Introduce the basic differences between hardware and software
- **Software Life-Cycles and Milestones**
  - Describe the variations of software life-cycles and how they fit into a project life-cycle
  - Introduce how life-cycle selection impacts a project
- **System Engineering Considerations**
  - Describe the relationship between system engineering and software engineering
  - Introduce the role of software system engineer/architect in system engineering activities
  - Impacts of systems architecture on software processes/development and integration and test considerations

Course Topics - continued

- **Software Technologies**
  - Introduce a primer on software-specific jargon and state-of-the-practice software technologies, such as OOA/OOD and UML
  - Provide planning, managing, and risk consideration of using new technology
- **COTS and Reuse**
  - Discuss the impact of using COTS or inherited software on software/system architecture and visa-versa
  - Describe the impact of using COTS on development activities, costs, and risk
- **Software Acquisition**
  - Provide a criteria for how to decide to contract and types of contracting
  - Discuss issues and factors to considered when acquiring software
- **Software Evaluation and Quality**
  - Discuss what it takes to build quality software
  - Describe various quality and testing techniques
  - Discuss the cost and management of reviews, testing, and quality assurance

Course Topics - continued

- **JPL, NASA, and Industry - Resources and Standards**
  - Discuss JPL and NASA policies, standards, and guidelines
  - Discuss JPL Software Principles and DNP (develop new products) processes and standards
  - Provide map of JPL resources and software expertise
  - Provide information on NASA and industry working groups
  - Provide references for further information
- **Configuration Management (CM) and Documentation**
  - Provide guidelines for adjusting the amount and types of documentation to suit project needs
  - Discuss the need and role of CM as well as related tools and metrics
- **Staffing Considerations and Workforce Planning**
  - Discuss software staffing structure throughout the development life-cycle, including the operational phase
  - Discuss software staff roles and responsibilities

Course Topics - continued

- **Understanding Software Planning and Metrics**
  - Provide information on rules-of-thumb for evaluating software plans and status
  - Discuss the need for planning and re-planning software development activities; levels of uncertainty throughout the life-cycle
  - Discuss metrics of interest to project management
- **Using the Mission Data System (MDS)**
  - Introduce what MDS is
  - Describe what is required to adapt MDS
- **Lessons Learned - Panel**
  - Provide a forum for discussion on software issues with project managers and practitioners
  - Provide Lessons Learned -- successes and common pitfalls

Current Plans

- **Current Activities**
  - selection of instructors
  - development and review of modules
- **Course Availability**
  - plan is to pilot course by the end of the fiscal year

2nd Annual GSFC-JPL Quality Mission Software Workshop

Relationships to Projects Software Quality Assurance

Quality Assurance Office (506)

Safety and Mission Assurance Directorate

Burton C. Sigal

May 17, 2000

Software Quality Assurance

- The Challenge
- Brief Overview of SQA Role
- JPL SQA & The NASA IV&V Facility
- Risk Driven Insight Program
- Risk Balance Profile
- Summary

The Challenge

- The amount of flight software being flown and the complexity of demands on that software are increasing dramatically, so it is becoming increasingly more important to...
- "...Do the right things right the 1st time…"
- Easy to say, but
  - How do we determine what are the 'right things' for a specific project?
  - How do we assure that they are done 'right'?
  - How do we get better at doing them?

Brief Overview of SQA Role

- **Match SQA tasks to key project drivers**
  - **What is right for this project?**
    - **We have work with the project to tailor, scale,…**
      - **Define “the right things”**
  - **Projects are unique**
    - **Time, risk, cost, functionality, performance, …**
  - **Projects are the same**
    - **Lessons learned**
    - **Regular and random**

Brief Overview of SQA Role

- **Ensure effective/correct results**
  - **How do we know things are correct?**
    - **We analyze, test, audit, … (verification and validation)**
      - **Static (early)**
      - **Dynamic (later)**
- **Don't do it all, just do what is critical/key**
- **Assess and recommend using Risk based analysis**
  - **Risk is a resource like schedule, cost, performance, etc., to be traded**

Brief Overview of SQA Role

- Software Risk Assessment
- Software Requirements Review
  - Requirements Analysis and Verification
  - Test Traceability Matrix
- Test Planning / Planning Assessment
- Software Validation and Verification
  - Integration Testing
  - System Level End-to-End Testing
  - Acceptance Testing/Systems Testing
  - Technical Status Reviews
- Test Tool Design and Development

Brief Overview of SQA Role

- Contractor/Partner/Supplier Insight Monitoring
- Process Engineering Support
- Design, Safety & Hazards Analysis
  - SFMECA
  - SFTA
  - Fault Protection
- Technology Infusion/RTOP
- Configuration Management
- Y2K Testing, Verification and Validation

SQA "Resume" 5/02/2000

Project	SQA Contact	Customer	Role	Service/Product	Result/Lessons Learned
26M Automation Task	Mikulski, C	Jeff Osman	Task Mgr.	Customer insight monitoring; S/W development products review/CDRs; Requirements Traceability of SOW to FRD to design.	
ACR	Schneider, Frank	Ben Parvin, Martha Berg	Task Manager, Software Mgr.	Subsystem Integration and Test Plan for DSCC Antenna Mechanical Subsystem Volume I Requirements (829-2); Reliability Analysis Feasibility for DSCC Antenna Mechanical System Controller; Formal Inspections; Software Product Assurance Play for 34M BWG Antenna; Test Trace Matrix	First usage of old data on analogous antenna system to predict what reliability and availability had to be to meet functional requirements for the new system design and functionality; Used detailed trac matrix to tie together 34M BWG subsystems for all internal antenna subsystems – HVAC; Monitor and Control, etc.
ACR	Wang, Monica	Ben Parvin	Task Mgr.	Subsystem Level Testing; Test Plans, Procedures Trace Matrix	
ACS (Advanced Communication Service)	Lam, Margaret	Brian Hammer	Proj Mgr.	System, Y2K Level Testing (S/W Test Plan, Procedures and Test Results)	
APC Upgrade	Mikulski, C	Ben Parvin/ Martha Berg	Task Mgr.	Generated & Managed the for FRD & SRD Requirements database; Reqs/Design Analysis & Trace Matrix; Test Plans, Procedures, Trace Matrix; Test witnessing; Development defect collection and evaluation.	
BVR	Lee, Susan Schneider, Frank	Ernest Stone	Task Manager	Formal Inspections	
Cassini	Lutz, Robyn	T Gavin & C Jones Sarah Gavit		Fault Protection; PVS Formal Methods	Formal specs and analysis allowed faster "red-flagging" of design anomalies
Cassini	Mathews, V	Chris Jones	S/C S/W Dev Mgr	S/W Library setup, CM, distribution; PFR track/close	
Cassini	Schneider, Frank	Chris Jones	Spacecraft Orbital Manager	Task 2 Critical Sequence Rollback Analysis	Model Checking was suggested as the preferred method to validate Cassini critical sequence rollback scheme

JPL SQA & The NASA IV&V Facility

- **June '99 direction from Administrator Goldin '...to better integrate IV&V activity into flight project activity…'**
- **JPL's response was the Project Software Quality Assurance Requirement (DMIE-44452) which states: "…The Software Quality Assurance organization shall perform an assessment and recommend for the projects/tasks, the appropriate level and mix of Software Quality Assurance and/or NASA IV&V Facility activities in support of the mitigation of safety and mission success risks associated with the project/task software…" .**
- **Several tools have been developed to support S/W risk management activities**
  - **Risk Driven (Partner) Insight Program**
  - **Risk Balance Profile (RPB)**
- **An assessment procedure is being developed to help identify SQA / IV&V activities focused on project specific S/W risk issues**

Risk Driven Partner Insight Program

- The Risk Driven Insight Program is a technique to help project managers and QA personnel assess and track ongoing performance of a multi-partner project.
- Features:
  - A qualitative insight approach rather than traditional oversight
  - Non-obtrusive to contractors / partners
  - Helps track ongoing performance / conformance of a multi-contractor project, with each contractor using their internal processes and documentation
  - Provides ongoing verification and monitoring at each phase and/or deliverable over the project life cycle
  - Directly extensible to internal S/W developments

Information Assessment Metrics

1.0 Software Management	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
1.1 How will the project/contractor (P/C) define a software development methodology and the software development life cycle phases?	-P458527 - MAP Sec. 5.0 applies MIL-Q-9858 and ISO - 9000 - 3 -P458682 - SDP Sec. 3.3 Sec. 4.2.1 Sec. 5.4..3	-674-XXX-200 Sec. 3.7 Sec. 3.7.1.2 -BATC S/W Eng. Man.(SEM) -Rapid Prototyping - Sys. Req. -XXX-IN0096-107 SDP Sec. 4.2.1, Sec 4.4.2.8	-CM#: XXX-SPEC-303-001 App. 10.0 SQMS - ANSI/ASQC Q 9001 Ref. ANSI/ASQC Q 9000-3	X		
1.2 How will the P/C manage the software development such that the deliverable software has a controlled development process?	-P458527 - MAP Sec, 4,4,6 PDR, CDR -P458682 - SDP (Dependency Diagram) Sec. 4.1 Sec. 5.1	-674-XXX-200 Sec. 3.7 Multiple baseline approach -XXX-IN0096-107 SDP Sec. 4.11, 4.1.2,4.1.3,4.2,4.2.2,4.3, 4.4.2.7 (SPF)	-CM#: XXX-SPEC-303-001 Joint Review - formal review SRR,PDR,CDR,TRR	X		
1.3 How will the P/C manage the software development such that the deliverable software products meets schedule and budget?	-P458682 - SDP Sec. 5.1 Sec. 5.1.2 Sec. 5.1.4.1 Sec. 5.2	-674-XXX-200 Sec. 3.7.1.1 A thorough req. analysis effort Sec. 3.7.1.6 -XXX-IN0096-107 SDP Sec. 4.2.1 Sec. 4.4.2.8	-CM#: XXX-SPEC-303-001 Sec. 1.9 DID	X		
1.4 What process will the P/C use to identify the required documents, and the type of review that the documents are subjected to?	-P458682 - SDP Sec. 4.1/A.3 Sec. 5.2	-674-XXX-200 Sec. 3.7, 3.7.1.1 S/W Develop Plan, S/W Req. Rev. (SRR) Sec. 3.7.2 Sec. 3.7.1.2 TLDR's -XXX-IN0096-107 SDP Sec. 4.3.1.1, 4.5.X	-CM#: XXX-SPEC-303-001 Sec. 2.3.2.2 DRT	X		
1.5 What specific processes will the P/C use to control S/W development in connection with H/W development phase and with regard to ECR's and mission success needs?	-P458682 - SDP Sec. 5.2	-674-XXX-200 Sec. 3.7 EVU -XXX-IN0096-107 SDP Sec. 4.3		X		

Information Assessment Metrics

2.0 Software Requirements and Software Design	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
2.1 How will the P/C perform the verification of those software requirements (including fault protection) are complete and consistent with mission needs?	-P458527 - MAP Sec.4.4.6 VCRM - Verification. Cross Ref. Matrix	-674-XXX-200 Sec. 3.8.1 -XXX-IN0096-107 SDP Sec. 4.3.1.1 Requirement Analysis	-CM#: XXX-SPEC-303-001 Sec. 2.4 PDR, CDR Sec. 3.1.1.2 Sys. Performance Verification Matrix	X		
2.2 How will the P/C perform the verification that the software design meets software requirements?	-P458527 - MAP Sec.4.4.7 -P458682 - SDP Sec. 5.2 Sec. 5.3	-674-XXX-200 Sec. 3.7.1.2 Case Tool, TLDRs, PDR, DDR, CDR CSCI+CSC →CSU using PDL Sec. 3.7.1.3 ICB -XXX-IN0096-107 SDP Sec. 4.3.1.2 Rapid prototype, Req. Matrices Sec. 4.5.2.1 Top Level DR, $.5.2.2 DDR	-CM#: XXX-SPEC-303-001 Sec. 2.3.1 DRP Sec. 2.3.2.2 DRT, SCR, PDR Sec. 10.2 GFE		X	
2.3 What type of measure will the P/C use to define all performance requirements?	-P458527 - MAP Sec.4.5 Document & Data control CMP, Inspection process, PA verifies.	-674-XXX-200 Sec. 3.7.1.1 CSDI -XXX-IN0096-107 SDP Sec 4.3.1.5 FQT	-CM#: XXX-SPEC-303-001 Sec. 3.0 EVS-SE. Sec. 3.1.1 Sys. Perf. Ver. Plan Sec. 10.2 GFE	X		
2.4 What type of process will the P/C use to define all interface requirements between hardware to software and software to software?	-P458527 - MAP Sec. XXXX- XXX Implementation Plan -P458682 - SDP Sec. 4.2.1, Sec. 5.2.3 (no req.) PR, Test, Static, Dynamic	-674-XXX-200 SEC 3.7.1.4 CSC & CSCI, EVU -XXX-IN0096-107 SDP Sec 4.3.1.4 CSC/CSCI Test	-CM#: XXX-SPEC-303-001 Sec. 3.1.1.1 Env. Ver. Plan Sec. 10.2 GFE	X		
2.5 What process will the P/C use to identify the design and implementation constraints?	-P458527 - MAP Sec.4.4.7 Master Plan, subsystem test plan and SDP	-XXX-IN0096-107 SDP Sec. 4.5.1.3 CDR Sec. 4.5.2 Informal Review Sec. 4.5.2.1 Top-level Design Rev. Sec. 4.5.2.2 Detailed Design Rev. Sec. 4.7.2 CM Principles (No specific process, comb. of above)			X	
2.6 What type of methods will the P/C use to specify how to detect/respond/recover the loss of critical function?	-P458682 - SDP Sec. 5.2.6.2 Sec. 5.4.1	-XXX-IN0096-107 SDP Sec. 4.7.3 Req. Change Request S/W Change Request Development Change Request		X		

Information Assessment Metrics

3.0 Choosing the Optimal Software Development Standard /Software Coding Convention and Code Maintainability	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
3.1 Which process will the P/C use to define the selected standards to implement the software?	-P458682 - SDP Sec. 4.2.2	-674-XXX-200 Sec. 3.7 - XXXXXX program -XXX-IN0096-107 SDP Sec 4.6.1 System flow Down Sec 4.6.2	-CM#: XXX-SPEC-303-001 Sec. 10.2 GFE		X	
3.2 What process will the P/C use to train personnel in the use of the standards and tools??	-P458527 - MAP Sec.4.1.8 ISO - 9001 IPT, SPC -P458682 - SDP Sec. 5.1.2.6	-674-XXX-200 Sec. 3.7.3 XXXXX FSW			X	
3.3 Which standard will the P/C use to define the software coding convention and standards that will be implemented?	-P458682 - SDP Sec. 4.2.2 Sec. 5.3.4	-674-XXX-200 Sec. 3.7.1.3 Inform. Code Rev. Sec. 3.7.3.1 SER→Case Tool/PDL -XXX-IN0096-107 SDP Sec. 4.6.2			X	
3.4 What process will the P/C use to perform the verification that the software coding convention and standards were appropriately applied?	-P458682 - SDP Sec. 4.2.2 Sec. 5.3.5	-674-XXX-200 Sec. 3.7.1.3 Eng. Verif. Unit (EVU), Logic Analyzer, Oscilloscope CSU,CSC integration Sec. 3.7.3.2 IR, CM before CSU "Assembly + C language" -XXX-IN0096-107 SDP Sec. 4.3..1.3			X	
3.5 What criteria will the P/C use to define code maintainability and adaptability for future upgrade?	-P458682 - SDP Sec. 4.2.3.1 XXXS98 Sec. 5.2.6.2	-674-XXX-200 Sec. 3.7.1.6 -XXX-IN0096-107 SDP Sec. 4.3.1.6 Sustaining S/W Eng.			X	

Information Assessment Metrics

4.0 Software Test Verification and Validation	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
4.1 What process will the P/C use to define a verification methodology or procedures to verify the software test plans and test procedures? (The test plans and procedures should be accurate and sufficient to check out the software and project requirements.)	-P458682 - SDP Sec. 5.2 Sec. 5.2.3	-674-XXX-200 Sec 3.7.1.4 CSS & CSCI Sec. 3.7.1.5 FQT Sec. 3.8 Early verif. End to End Complete Model -XXX-IN0096-107 SDP Sec. 4.5.2.4 Integrated Ready Review		X		
4.2 What are the criteria definitions will the P/C use to determine that the software is ready to proceed into the Acceptance Test phase?	-P458682 - SDP Sec. 5.2.3.5 Sec. 5.3.6 Sec. 5.3.6.2	-XXX-IN0096-107 SDP Sec. 4.5.2.5 Test Readiness Review		X		
4.3 Will the P/C have a defined verification matrix to verify the test procedures are being carried out correctly?	-P458682 - SDP Sec. 5.3	-674-XXX-200 Sec. 3.8 Self Test Capability Sec. 3.8.1 Protoflight Philosophy Sec. 3.8.1.1 Integration of SI -XXX-IN0096-107 SDP Sec. FSW Test Plan		X		
4.4 How will the P/C verify the that actual test results are correctly checked against expected results?	-P458682 - SDP Sec. 5.4.1 Sec. 5.3.6.3	-XXX-IN0096-107 SDP Sec. 4.5.2.6 Test Results Review		X		
4.5 Will the P/C have a defined process to assure test anomalies or defects are accurately recorded and reported?	-P458682 - SDP Sec. 5.4.1 Sec. 5.3.6.3 Sec. 5.4.1	-XXX-IN0096-107 SDP Sec. 4.4.2.9 Change Request		X		
4.6 What will be the process pathway that the P/C will use to validate the robustness of the S/W with regards to non-nominal status, timing margin, through out the mission needs and bandwidth? (This should be covered within the test procedures).	-P458682 - SDP Sec. 5.1.4.1 Observatory level Sec. 5.2 Sec. 5.2.4 Sec. 5.2.5	-674-XXX-200 Sec. 3.7.1.1 Timing and Sizing assessment Sec. 3.7.1.2 Sec. 3.7.1.4 -XXX-IN0096-107 SDP Sec. 4.3.1.2 Program Design Lang.			X	

Information Assessment Metrics

	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
4.7 How will the P/C verify that the deliverable software has met all software system requirements with exception of the specified liens?		-674-XXX-200 Sec. 3.7.1.5 FQT, SRD		X		
4.8 What type of criteria-definitions that the P/C will use to accept third party software (if applicable)?	-P458527 - MAP Sec.3.0 ISO-9001, ISO-9003 GIDEP, ALERT	-XXX-IN0096-107 SDP Sec. 4.7.1.1 Purchased S/W				X
5.0 S/W Development Tools						
5.1 What is the process pathway will the P/C use to assure a proper development environment?	-P458682 - SDP Sec. 5.4.1	-XXX-IN0096-107 SDP Sec 4.3 FSW Dev. Process				X
5.2 How will the P/C assure the evolution of software tool support will not become obsolete throughout the project and continue support after post launch?	-P458682 - SDP Sec. 5.1.3.3	-XXX-IN0096-107 SDP Sec. 4.7.2 S/W Product CM (indirectly)				X
6.0 S/W Problem Reporting/Resolution						
6.1 What is the process pathway will the P/C use to manage, identify, track, and verify problem failure reporting?	-P458682 - SDP Sec. 5.4.1	-674-XXX-200 Sec. 3.7.5 -XXX-IN0096-107 SDP Sec. 4.4.2.10 S/W input to MMR (weak)		X		
6.2 What is the process pathway will the P/C use for corrective action of software problems and implementation of software changes? The process should also include anomalies with and without hardware induction.	-P458682 - SDP Sec. 5.4.1 Sec. 5.4.2	-674-XXX-200 Sec. 3.7.5 -XXX-IN0096-107 SDP Sec. 4.4.2.9 Change Request		X		

Information Assessment Metrics

7.0 Software Documentation	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
7.1 Will the P/C have a defined traceability process or tool to verify that the contents of each software documents are accurate, consistent, and properly reflecting the purpose of the corresponding document and the project requirements?	-P458682 - SDP Sec. 5.2 Sec. 5.3.2 Sec. 5.3.1.6	-674-XXX-200 Sec. 3.7.1.1 Req. Traceability Matrix XXX				X
8.0 Software Configuration Management (CM)						
8.1 What are the criteria will the P/C use to define the readiness for baseline of hardware, firmware, and software?	-P458682 - SDP Sec. 5.4.1.2 Sec. 5.4.3	-674-XXX-200 Sec 3.7 Multiple Baseline Control NIMOS program Sec. 3.7.2.1 SDP, SRD, SDD, UOM, SPLD, STP, STPR -XXX-IN0096-107 SDP Sec. 4.3, 4.7.2	-CM#: XXX-SPEC-303-001 Sec. 10.1.2 Corrective Action Process, Problem reporting	X		
8.2 What type of mechanism pathway will the P/C use for identifying, maintaining control, and tracking of all configuration items, their associated documentation, and any changes to them?	-P458682 - SDP Sec. 5.4.4	-674-XXX-200 Sec. 3.7.1.3 Sec. 3.7.4 SDL -XXX-IN0096-107 SDP Sec. 3.1 SUDF Sec. 4.7, 4.7.1 S/W Dev. Lib Sec. 4.7.2, 4.7.3 CM Principles	-CM#: XXX-SPEC-303-001 Sec. 10.1.2 Informal Control (SCM) Sec. 10.2 GFE - SQMS(20% for significant)	X		
9.0 Software Quality Control						
9.1 What are the document procedures will the P/C establish for identifying, collecting, accessing, storing, and disposing of quality records?	-P458527 - MAP Sec.4.15, 4.16 -P458682 - SDP Sec. 4.2.7	-XXX-IN0096-107 SDP Sec. 4.7, 4.7.1 S/W Dev. Lib Sec. 4.7.2, 4.7.3 CM Principles				X
9.2 What are the metric-definitions will the P/C use to reflect unfavorable trends on schedules and the development phases?	-P458527 - MAP Sec.4.13 -P458682 - SDP Sec. 5.1.4.1	-XXX-IN0096-107 SDP Sec. 4.7.1 SDL Sec. 4.7.2, 4.7.3	-CM#: XXX-SPEC-303-001 Sec. 10.1.3 SCM classifications	X		
9.3 What are the established and maintained procedures will the P/C use for validation, storage, protection and maintenance of configuration items used by the P/C (including third parties products, if applicable)?	-P458527 - MAP Sec. 1.0 ISO-9001:1994	-XXX-IN0096-107 SDP Sec. 4.7.2 CM Principles				X

Information Assessment Metrics

10.0 Software Fault Protection	CONTRACTOR A	CONTRACTOR B	CONTRACTOR C	Priority High	Med.	Low
10.1 Does the P/C have a "fault" definition requirement including its criticality and impact to S/C operation?	-FP CDR RAR 7.0-3 Fault Protection Requirements Fault Classification -FP CDR SI RAR7.0-1—3 Key Req.	-FP CDR-XXX:10-13 Fault Protection Requirements -FP CDR-XX/XX/XXX F90011-Sec 4-6 FP Requirements	-FP CDR XXX Key FP Requirements (FPRD:Sec.4.4-4.4.1)	X		
10.2 Does the P/C have a process to detect fault?	-FP CDR FSW/CCDHS RAR 7.0-5---23 -FP CDR PCS4 RAR 7.0---9 -FP CDR…..	-FP CDR-XXX:10-13 Implementation Approach, Credible Failure Mode -FP CDR- XX/XX/XXX F90011-Sec 11 Op. State	-FP CDR XXX WEA Sensor Monitoring, Sensor Faults	X		
10.3 How will the P/C ensure that the responding mechanism to handle fault is efficient?	-FP CDR System RAR 7.0-2—3 Uplink, Downlink -FP CDR System 12 RAR 7.0-10-17	-FP CDR-XXX:14-17 Success Criteria's F90011-Sec 19 Error Handle F90011-sec 21 Verified by Test	-FP CDR XXX- Process Faults			
10.4 What type of mechanism pathway will the P/C use for managing the recovery of a single system element from a fault condition without affecting the normal operation of other subsystems in the S/C and observatory?	-FP CDR System RAR 7.0-2—3	-FP CDR-XXX:14-17 Mechanism Operation -FP CDR-F90011-Sec 12 FP States	-FP CDR FMEA Status	X		
10.5 What process does the P/C have to ensure that if a single subsystem cannot clear a fault, will the subsystem be placed in a safe and commandable state that can be maintained until SC or ground corrective action arrives?	-FP CDR System RAR 7.0-4—5 State Mode	-FP CDR-XXX:15 Protection -FP CDR-XXX:14 State Transition -FP CDR CD F90011-Sec-8 Design Change -FP CDR CD F90011-Sec-12 FP State -FP CDR CD F90011-Sec-20 Fault Recovery	-FP CDR FMEA Status	X		
10.6 How will the P/C ensure that a single subsystem failure is protected from propagating failures to any internal or external subsystems of the S/C?		-FP CDR-XXX:15 Protection -FP CDR-XXX:38-41 CTA-Single point failure	-FP CDR XXX Autonomous FP Criteria	X		
10.7 How will the P/C handle fault messages from S/C to Ground?		-FP CDR-XXX:16-17 Mechanism Operation -FP CDR CD F90011-Sec-13-15 C&DH Monitors CE and Responds to Faults	-FP CDR S/C Responses to XXX Faults	X		

Information Assessment Metrics

11.0 Software Safety	CONTRACTOR A			CONTRACTOR B			CONTRACTOR C			Priority High	Med.	Low
11.1 How will the P/C ensure that systems safety is maintained and that spacecraft system hardware and software are designed to provide protection against all points of failure?	-P458682 - SDP Sec. 4.2.4									X		
11.2 How will the P/C ensure that the software specifications contain adequate safety axioms and requirements to assure safe operation of the system?	-P458682 - SDP Sec. 4.2.4						-CM#: XXX-SPEC-303-001 Sec. 10.3 will conduct NSS 1740.13 if needed			X		
11.3 How will the P/C ensure that subcontractor or government agency supplied software meets requirements for high integrity software and can be integrated into the whole system with no negative impact on the safety of the system?										X		
PRIORITIES	High	Med.	Low	High	Med.	Low	High	Med.	Low	High	Med	Low
Total # of Identifiable Objectives(O)												
*Multiply by Priority Weights (W)	x9	x3	x1	x9	x3	x1	x9	x3	x1	x9	x3	x1
Total weighted Identifiable Objectives (X=O*W)	=X1			=X2			=X3			$Y=\sum_{H,M,L}$ Total		
** % Information Assessment (X/Y)**	X1/SUM			X2/SUM			X3/SUM			= SUM		

Risk Driven Partner Insight Program

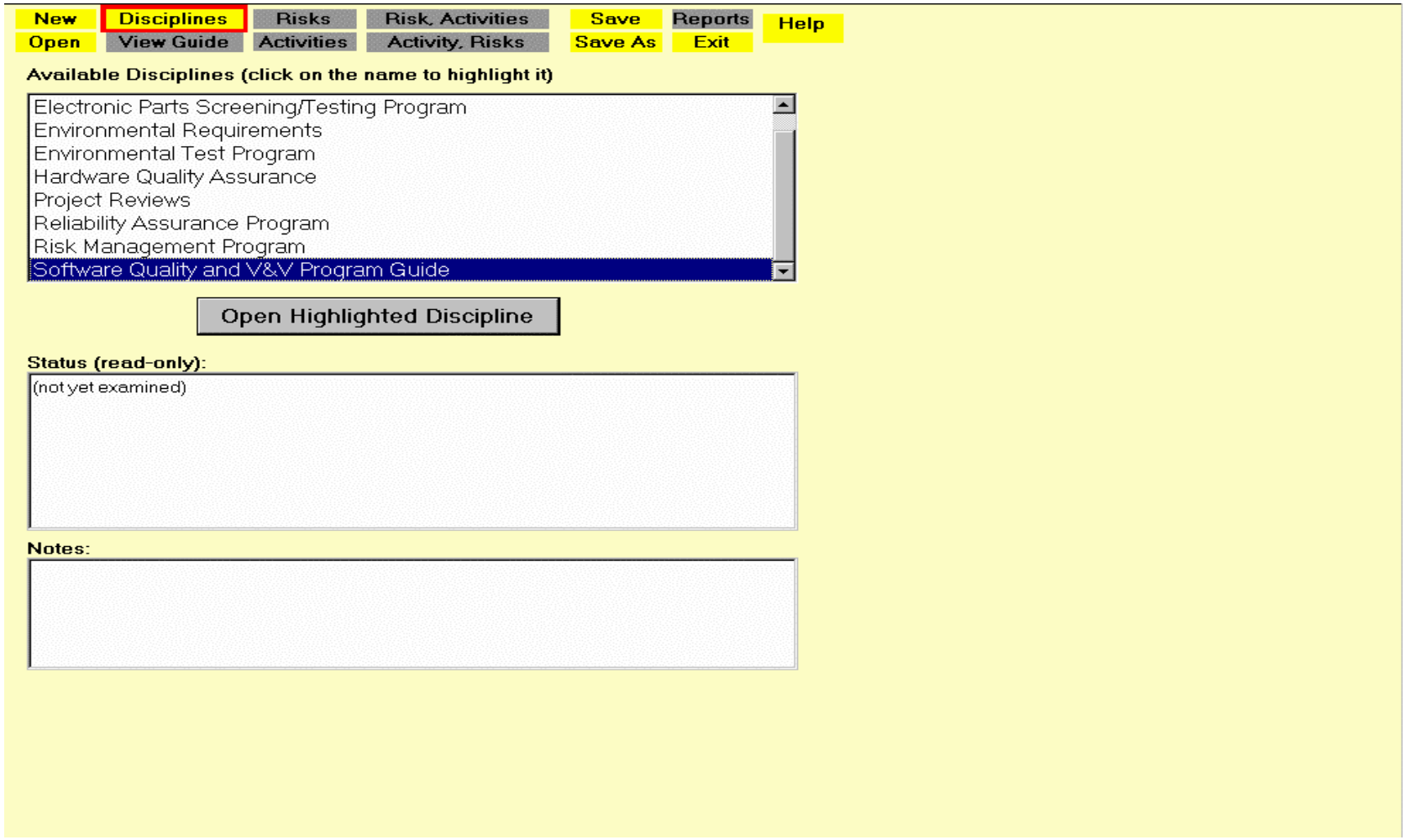
- Results:
  - Applied to several flight projects
  - Enabled project / mission assurance managers and QA personnel to focus on areas needing attention for mission success.
  - Facilitated conformance monitoring without using the traditional approach of imposing external standards and /or Data Item Descriptions (DIDs).
  - Groups of contractors working on the same project were able to employ their own viable quality processes, while the prime contractor (or sponsor) maintained sufficient insight
  - Projects are finding the process very value-adding and are extending its purview

The Risk Balancing Profile (RBP)*

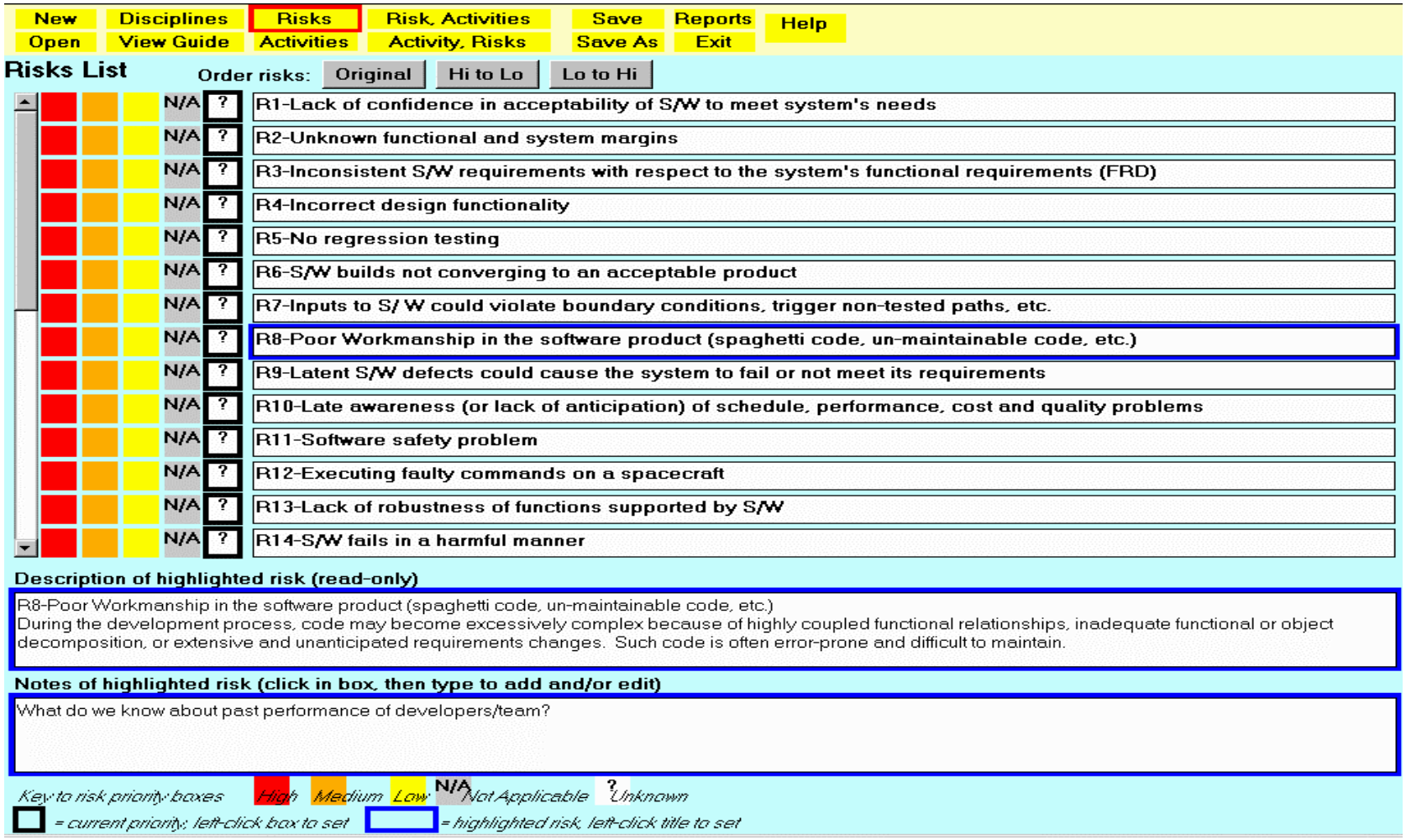
- A tool designed to aid in identifying project specific risks
  - Allows for assessing/assigning level of relative criticality
- Designed for (initial) use by non-domain experts
  - Supports analysis prior to consulting with domain experts
  - Supports ongoing consulting with domain experts
- Supports tailoring project content to project specific risks
  - Suggests contents for a minimum risk project
  - Suggests contents for a minimum content project
- Optimization strategy based on content/mitigation, lessons learned
- Enables what-if analysis
- Allows for updating over project life cycle

* Under Code Q sponsorship

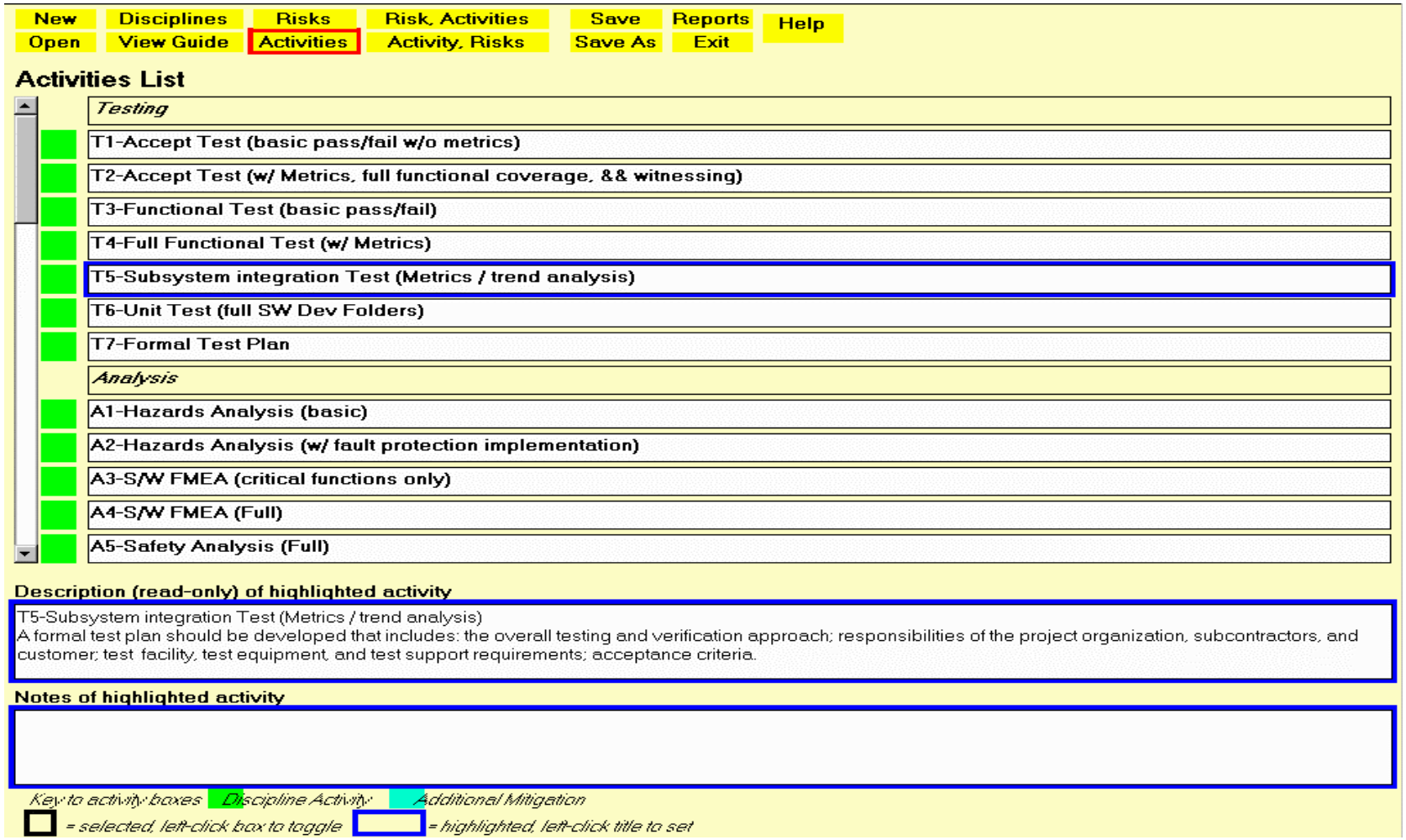
Opening Screen (Select a Discipline)



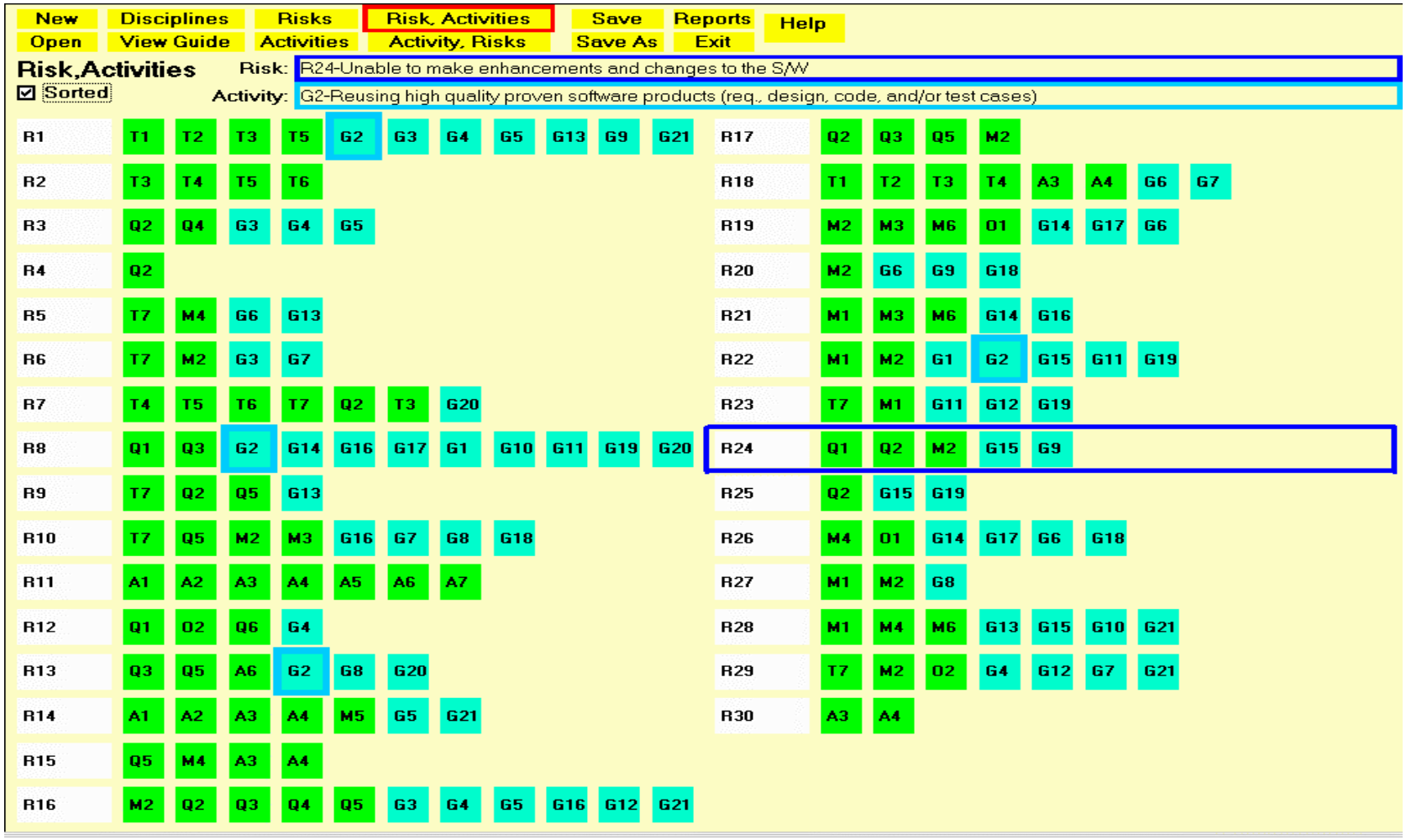
Initial Listing By Risk



Initial Listing By Activity



Risks and Their Activities



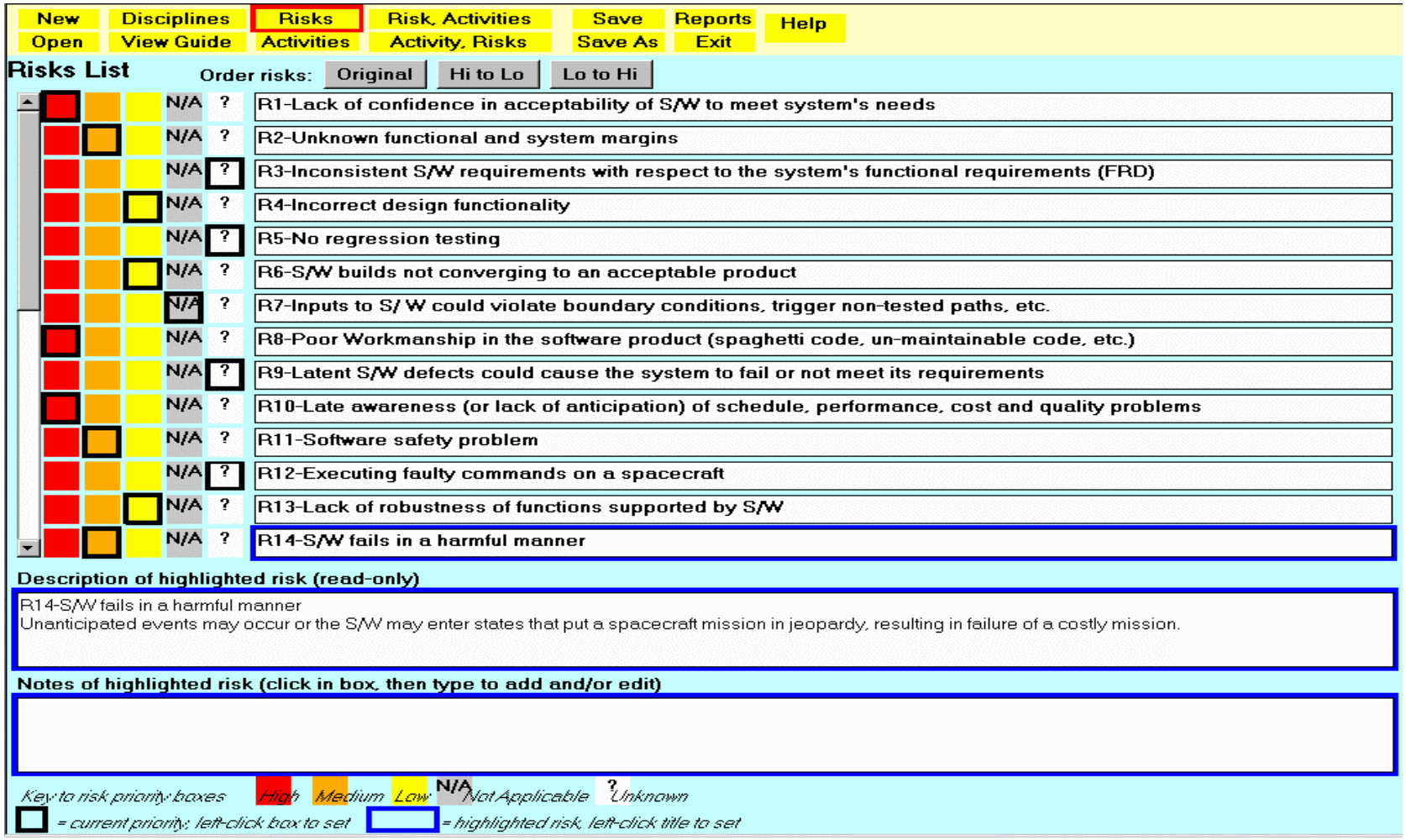
Activities and Their Risks

New | Disciplines | Risks | Risk, Activities | Save | Reports | Help
Open | View Guide | Activities | Activity, Risks | Save As | Exit

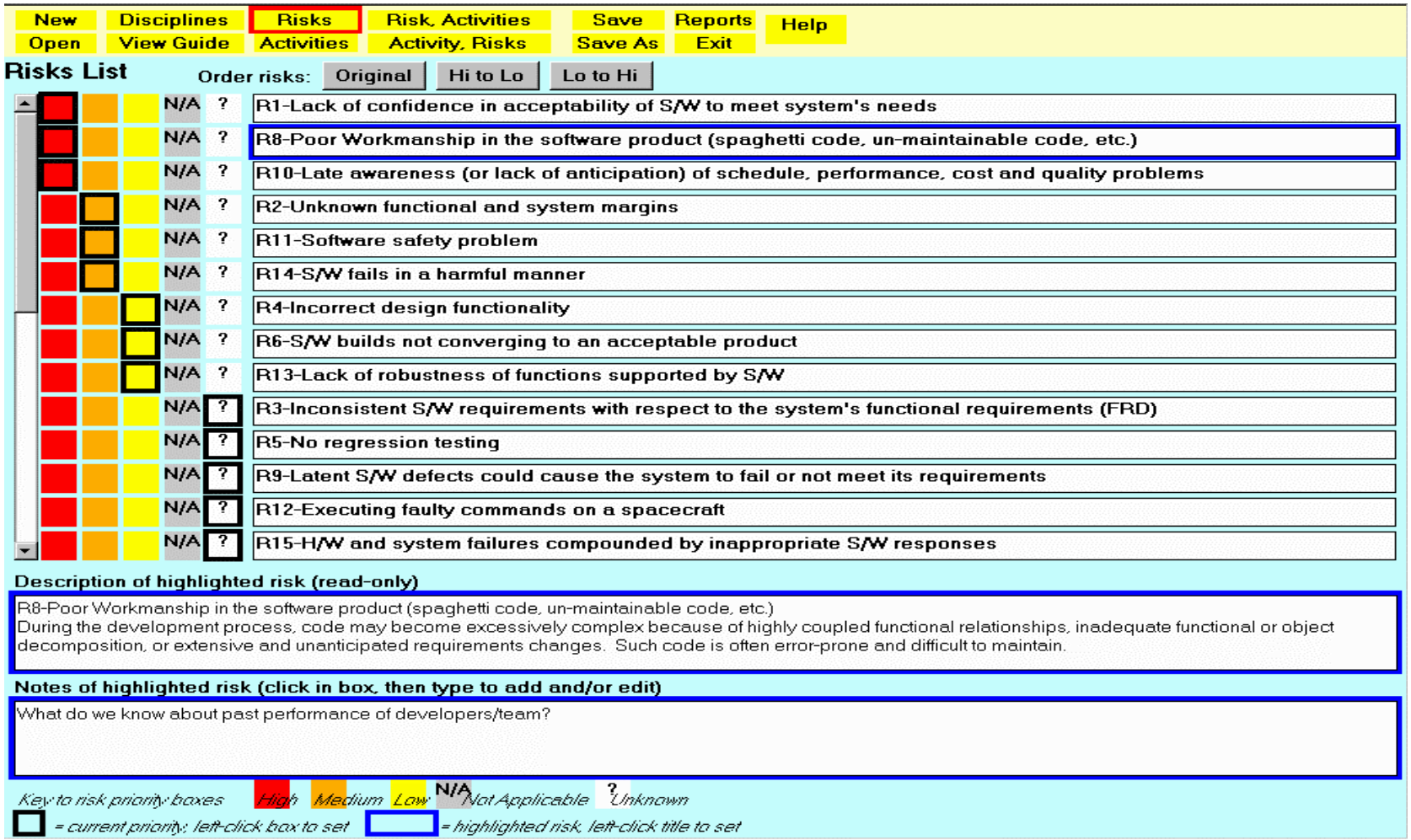
Activity,Risks Activity: T1-Accept Test (basic pass/fail w/o metrics)
Risk:

T1	R18	R1								
T2	R18	R1								
T3	R18	R1	R2	R7						
T4	R7	R18	R2							
T5	R7	R1	R2							
T6	R7	R2								
T7	R7	R9	R10	R5	R6	R23	R29			
A1	R11	R14								
A2	R11	R14								
A3	R11	R18	R14	R15	R30					
A4	R11	R18	R14	R15	R30					
A5	R11									
A6	R11	R13								
A7	R11									
Q1	R8	R12	R24							
Q2	R7	R9	R17	R3	R4	R16	R24	R25		
Q3	R8	R17	R13	R16						
Q4	R3	R16								
Q5	R9	R10	R17	R13	R15	R16				
Q6	R12									
M1	R21	R22	R23	R27	R28					
M2	R10	R16	R17	R19	R20	R6	R22	R24	R27	R29
M3	R10	R19	R21							
M4	R5	R15	R26	R28						
M5	R14									
M6	R19	R21	R28							
O1	R19	R26								
O2	R12	R29								
G1	R22	R8								
G2	R1	R8	R13	R22						
G3	R1	R3	R6	R16						
G4	R1	R3	R12	R16	R29					
G5	R1	R3	R14	R16						
G6	R5	R18	R19	R20	R26					
G7	R6	R10	R18	R29						
G8	R10	R13	R27							
G9	R1	R20	R24							
G10	R8	R28								
G11	R8	R22	R23							
G12	R29	R16	R23							
G13	R1	R5	R9	R28						
G14	R8	R19	R21	R26						
G15	R22	R24	R25	R28						
G16	R8	R10	R16	R21						
G17	R8	R19	R26							
G18	R10	R20	R26							
G19	R8	R22	R23	R25						
G20	R7	R8	R13							
G21	R1	R14	R16	R28	R29					

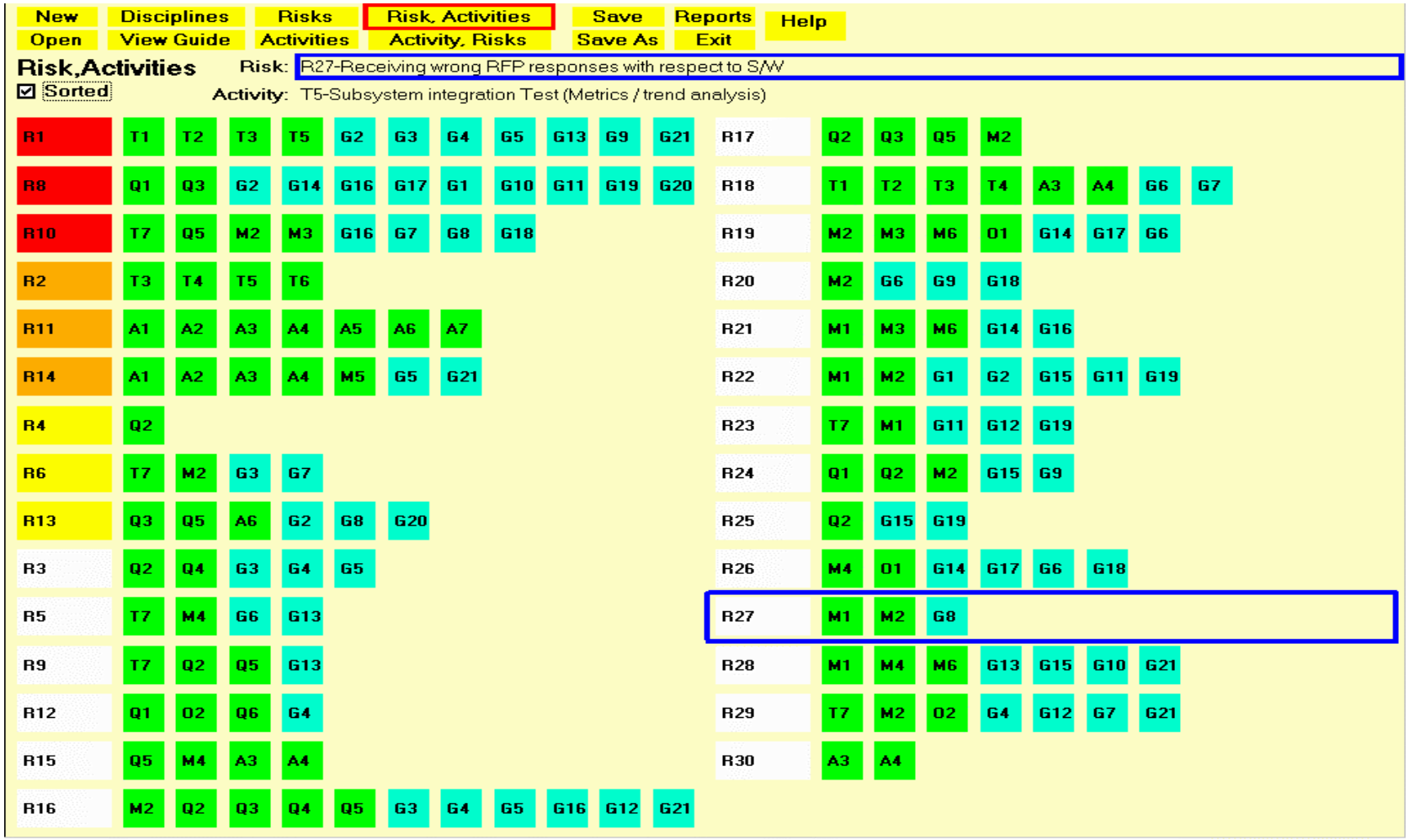
1st Cut At Defining Risk Criticality



1st Cut Sorted By Weighted Risk



Sorted Risk Activity



Activity, Weighted Risks

New | Disciplines | Risks | Risk, Activities | Save | Reports | Help
Open | View Guide | Activities | Activity, Risks | Save As | Exit

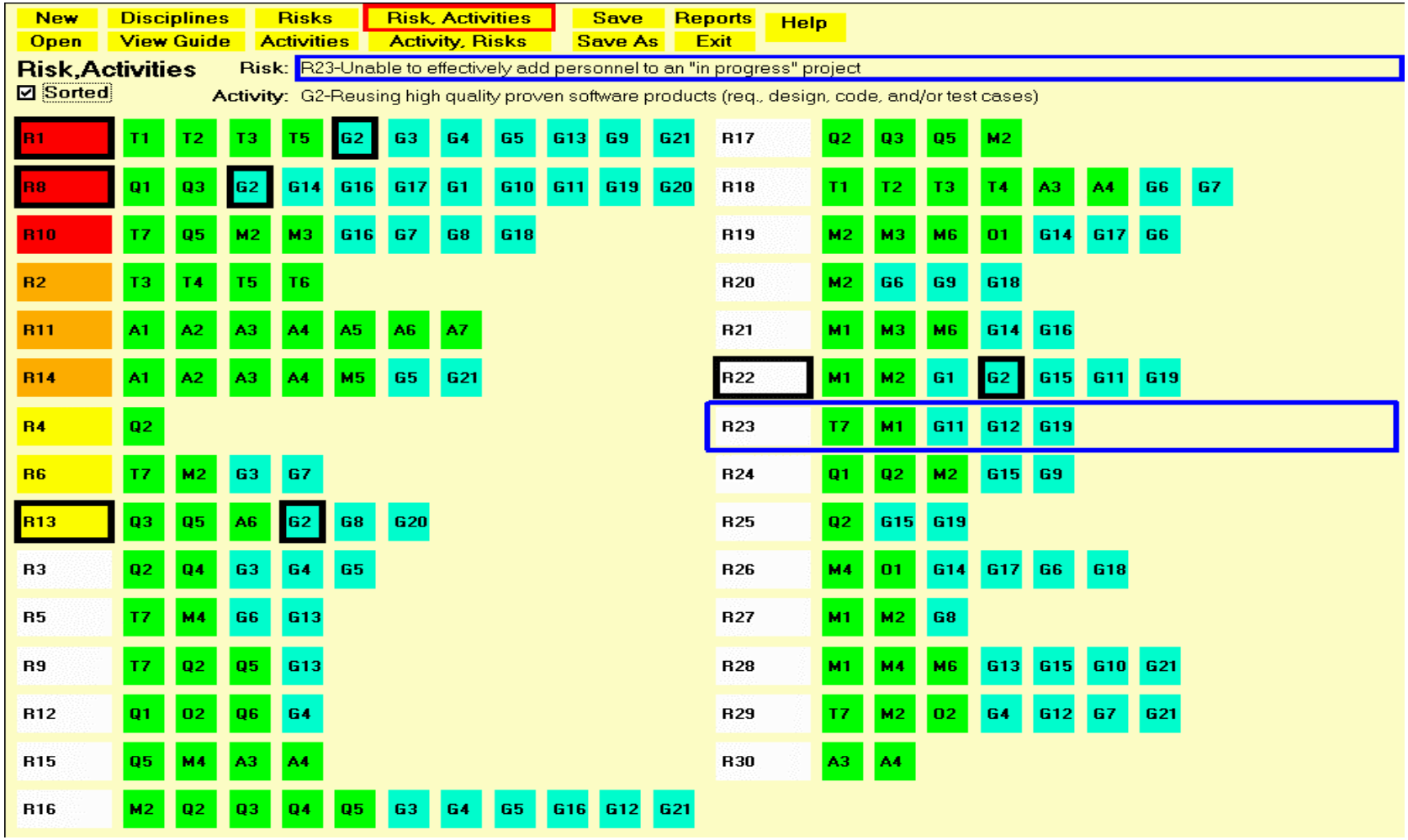
Activity,Risks Activity: G18-Use EVA metrics

Risk:

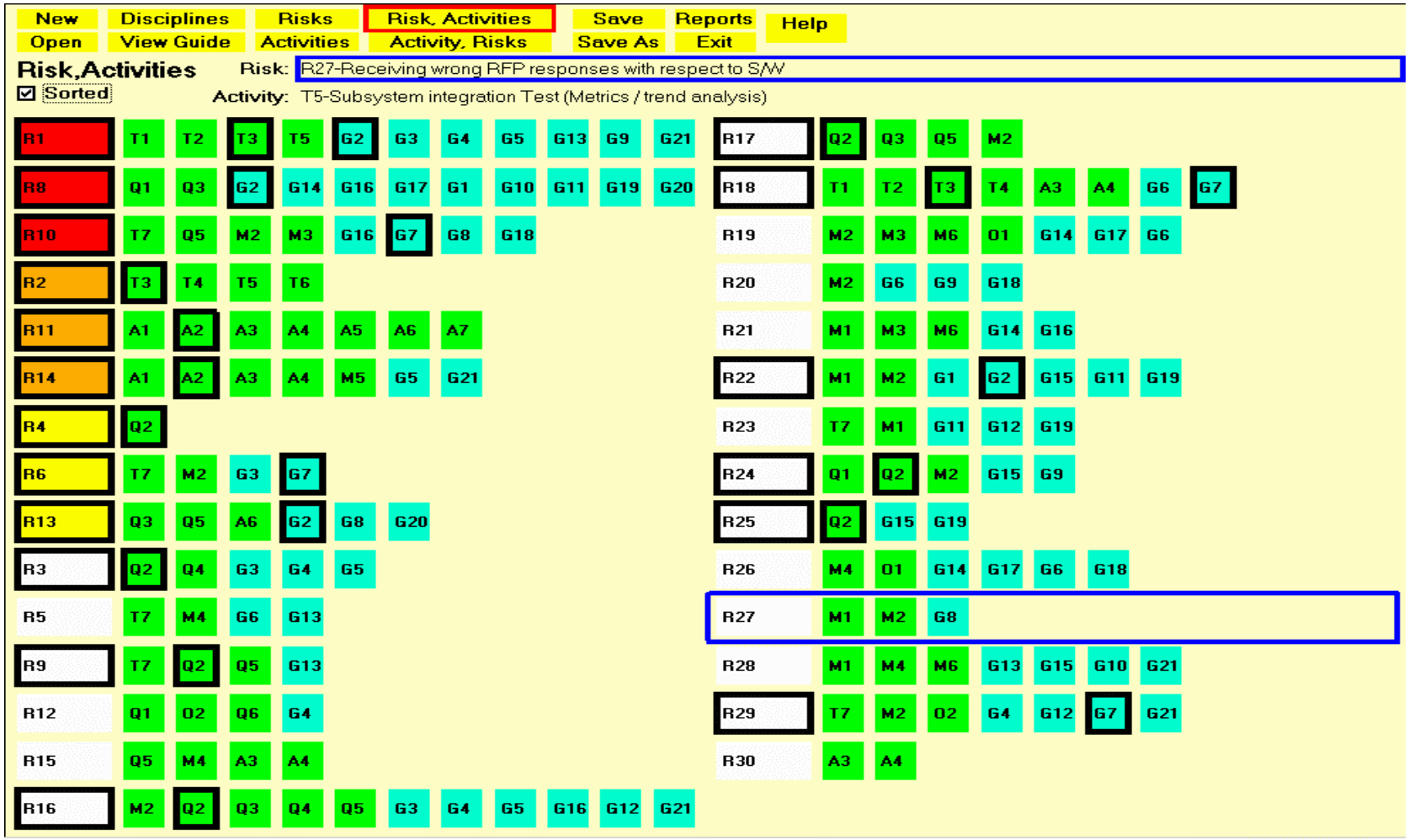
T1	R1	R18								
T2	R1	R18								
T3	R1	R2	R18							
T4	R2	R18								
T5	R1	R2								
T6	R2									
T7	R10	R6	R9	R5	R23	R29				
A1	R11	R14								
A2	R11	R14								
A3	R11	R14	R18	R15	R30					
A4	R11	R14	R18	R15	R30					
A5	R11									
A6	R11	R13								
A7	R11									
Q1	R8	R12	R24							
Q2	R4	R9	R17	R3	R16	R24	R25			
Q3	R8	R13	R17	R16						
Q4	R3	R16								
Q5	R10	R13	R9	R17	R15	R16				
Q6	R12									
M1	R21	R22	R23	R27	R28					
M2	R10	R6	R16	R17	R19	R20	R22	R24	R27	R29
M3	R10	R19	R21							
M4	R5	R15	R26	R28						
M5	R14									

M6	R19	R21	R28		
O1	R19	R26			
O2	R12	R29			
G1	R8	R22			
G2	R1	R8	R13	R22	
G3	R1	R6	R3	R16	
G4	R1	R3	R12	R16	R29
G5	R1	R14	R3	R16	
G6	R5	R18	R19	R20	R26
G7	R10	R6	R18	R29	
G8	R10	R13	R27		
G9	R1	R20	R24		
G10	R8	R28			
G11	R8	R22	R23		
G12	R29	R16	R23		
G13	R1	R5	R9	R28	
G14	R8	R19	R21	R26	
G15	R22	R24	R25	R28	
G16	R8	R10	R16	R21	
G17	R8	R19	R26		
G18	R10	R20	R26		
G19	R8	R22	R23	R25	
G20	R8	R13			
G21	R1	R14	R16	R28	R29

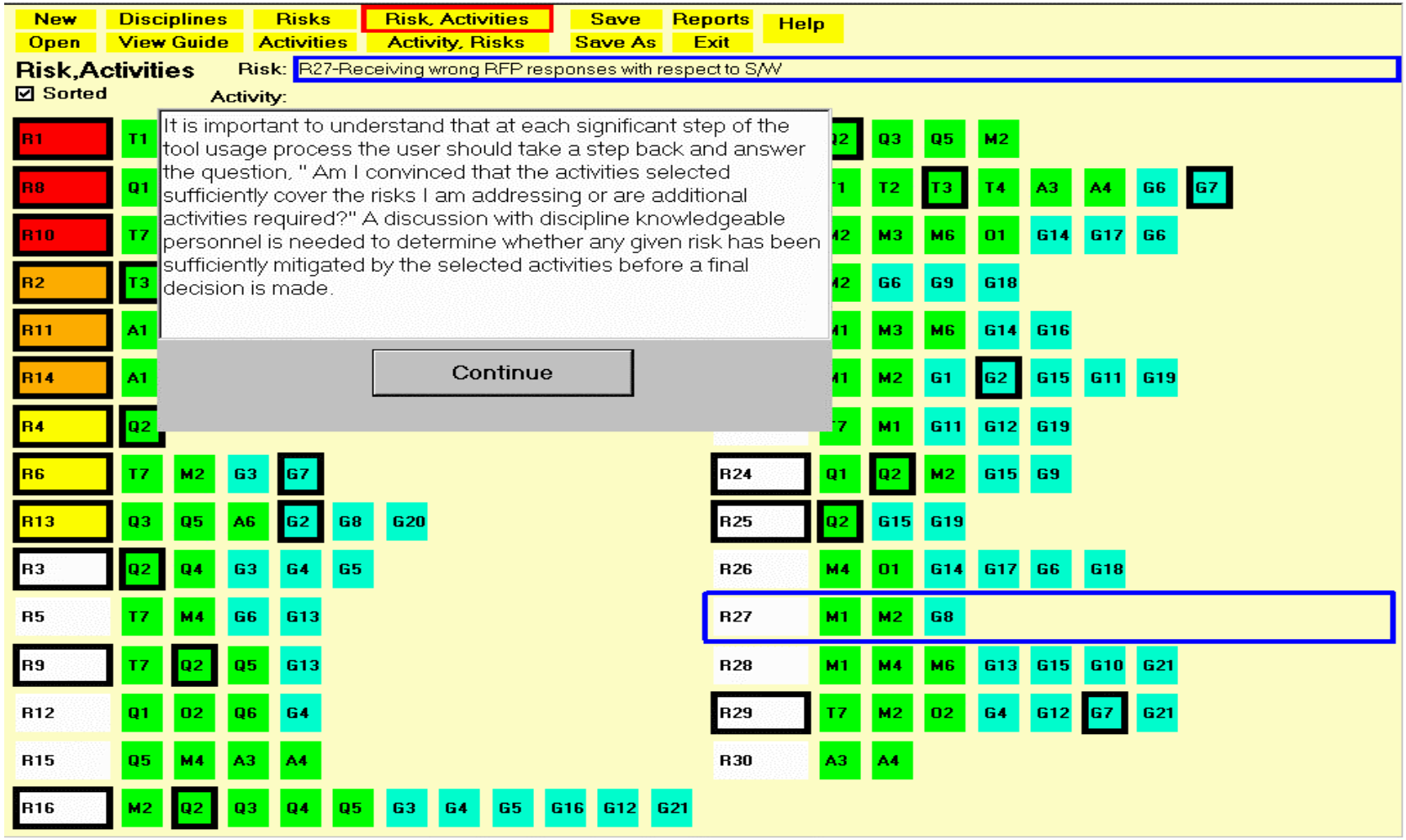
1st Cut At Assigning Strategy



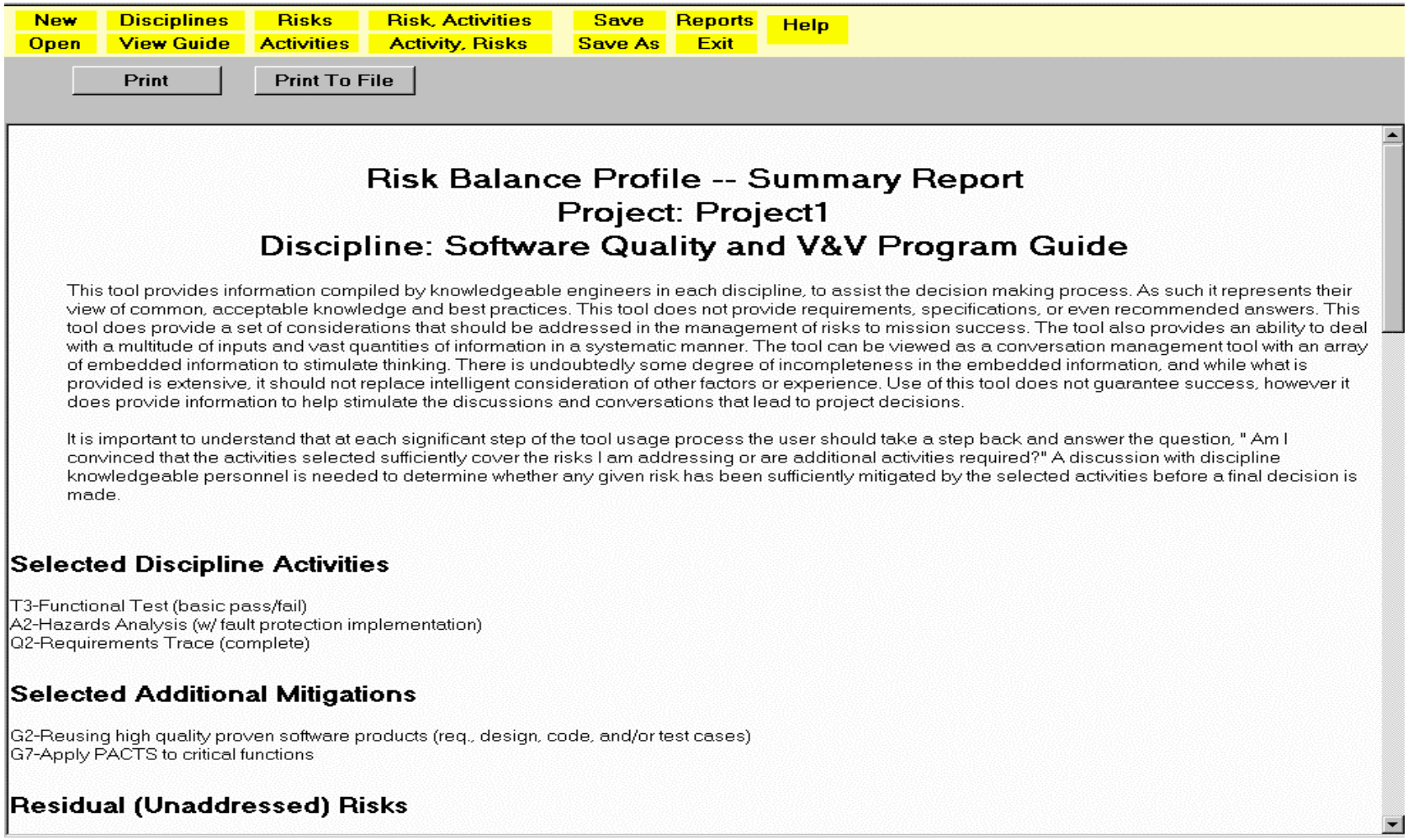
Final Cut of Activities by Risks



Are You Sure?


New
Disciplines
Risks
Risk, Activities
Save
Reports
Help
Open
View Guide
Activities
Activity, Risks
Save As
Exit
Risk,Activities
Risk: R27-Receiving wrong RFP responses with respect to S/W
Sorted
Activity:
It is important to understand that at each significant step of the tool usage process the user should take a step back and answer the question, " Am I convinced that the activities selected sufficiently cover the risks I am addressing or are additional activities required?" A discussion with discipline knowledgeable personnel is needed to determine whether any given risk has been sufficiently mitigated by the selected activities before a final decision is made.
Continue

Summary Report - 1



Summary Report - 2

New | Disciplines | Risks | Risk, Activities | Save | Reports | Help
Open | View Guide | Activities | Activity, Risks | Save As | Exit

Print | Print To File

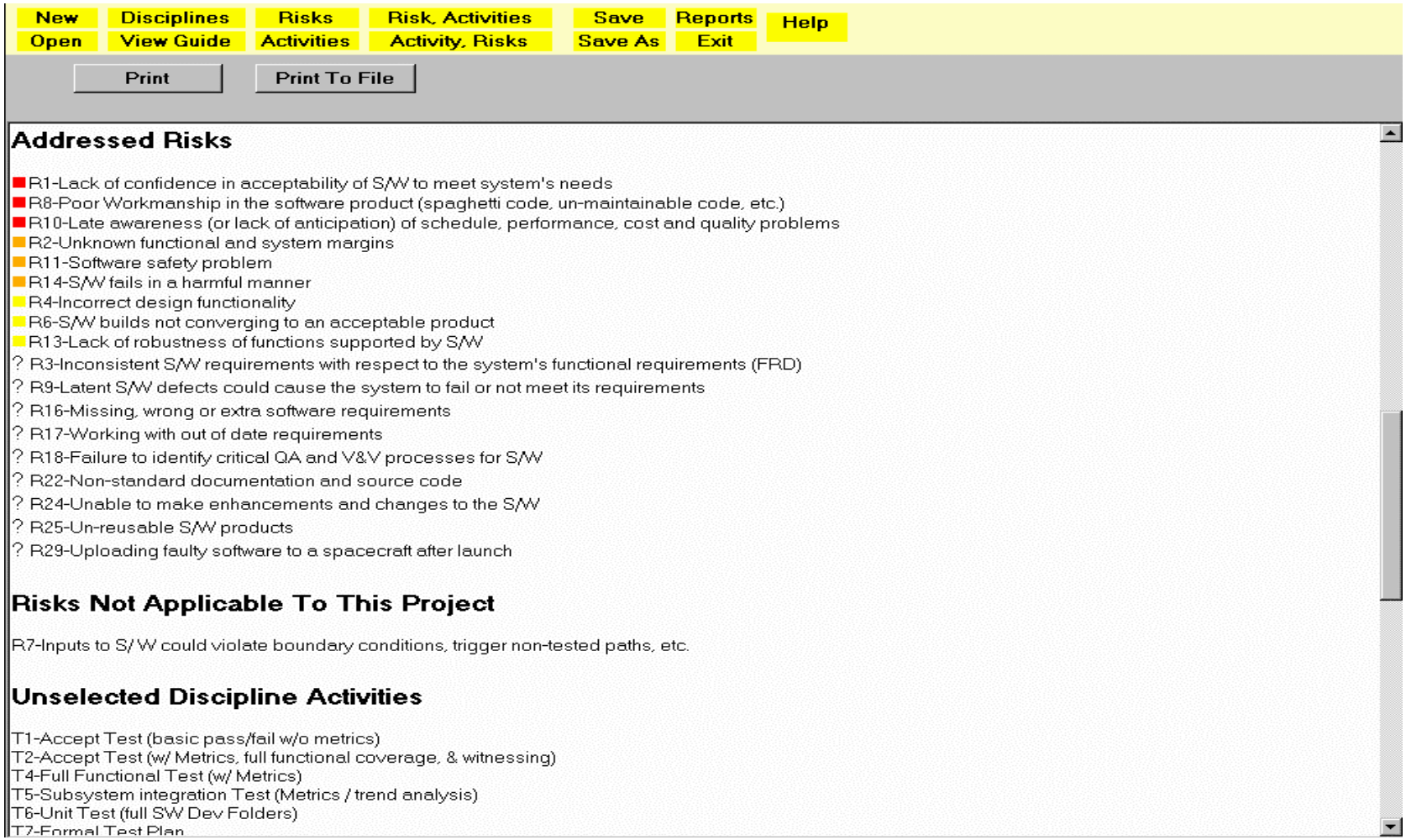
Residual (Unaddressed) Risks

? R5-No regression testing
? R12-Executing faulty commands on a spacecraft
? R15-H/W and system failures compounded by inappropriate S/W responses
? R19-Failure to identify critical contractor monitor points
? R20-Can't identify changes impacts (cost, schedule, functionality, etc.)
? R21-Project progressing to next phase of development before ready
? R23-Unable to effectively add personnel to an "in progress" project
? R26-Choosing the wrong/high risk contractor to develop software
? R27-Receiving wrong RFP responses with respect to S/W
? R28-Encountering a S/W error that wasn't tested
? R30-How software will behave when failures occur is unknown

Addressed Risks

R1-Lack of confidence in acceptability of S/W to meet system's needs
R8-Poor Workmanship in the software product (spaghetti code, un-maintainable code, etc.)
R10-Late awareness (or lack of anticipation) of schedule, performance, cost and quality problems
R2-Unknown functional and system margins
R11-Software safety problem
R14-S/W fails in a harmful manner
R4-Incorrect design functionality
R6-S/W builds not converging to an acceptable product
R13-Lack of robustness of functions supported by S/W
? R3-Inconsistent S/W requirements with respect to the system's functional requirements (FRD)
? R9-Latent S/W defects could cause the system to fail or not meet its requirements
? R16-Missing, wrong or extra software requirements
? R17-Working with out of date requirements
? R18-Failure to identify critical QA and V&V processes for S/W
? R22-Non-standard documentation and source code
? R24-Unable to make enhancements and changes to the S/W

Summary Report - 3



What the RBP Guide "Is" and "Is Not"

- The RBP Guide is:
  - Useful for identifying project risk associated with a selected level of SQA /V&V program content
  - Useful for identifying mitigation possibilities
  - Helpful in planning appropriate resources for QA / V&V program content (and balancing resources across various project risk reduction areas)
- The RBP Guide is not:
  - A substitute for an experts' participation during the planning process
  - Prescriptive in nature (it is intended to illustrate how to tailor a SQA / V&V program)
  - A process monitoring and corrective action technique
- There are no 100% certain, 0% Risk Programs

Goddard Space Flight Center

JPL

Summary

- The amount of flight software being flown and proposed for the new millennium and the complexity of demands on that software are increasing dramatically
- Meeting the quality demands of flight software requires new approaches to quality assurance to ensure a robust product within project constraints
- Treating project specific risks as a resource to be traded like other project resources offers an effective solution
- Risk-assessment based tools which are easy to use over the project life cycle and allow tailoring, iteration, updating, and provide lessons learned, are a key part of that solution

2nd Annual GSFC-JPL Quality Mission Software Workshop

Relationships to Projects
Software Development Principles

JPL

Milton L. Lavin

17 May 2000

Topics

- Principle Definition and Scope
- Motivation for Principles
- Intended Use
- Approach to Development
- Overview of Content
- Some Examples

Principle Definition and Scope

- **Principle: Fundamental best practice, proven to be effective in flight system development, to be deviated from only for sound reasons.**

- **Scope:**
  - **Emphasis on mission critical software**
  - **Formulation, design, implementation, and operations**
  - **Both management and design activities**
  - **Applies to subcontractors & partners**

Motivation for Principles

- **Continual cost overruns, schedule compression, expensive rework & defects found in operations indicate systemic problems (c.f. Cost/Risk Study)**
- **Software management expertise is spread over many small missions on tight budgets & schedules**
- **Faster/better/cheaper is here to stay; JPL cannot afford frequent failures**

Intended Use

- **Software principles will be integrated with JPL D-17868, the more general principles for flight systems.**
- **Project Implementation Plan and Software Development Plan will document compliance; no waivers are needed.**
- **Adherence will be verified in reviews of both JPL and out-of-house development.**

Approach to Development

- **Inclusion Criteria:**
  - **Make a difference in cost/schedule/quality**
  - **Relevant to JPL**
  - **Omit what is widely practiced**
  - **Applicable to wide range of projects**
  - **Useful to PM/PEM, Reviewers, Practitioners**
- **Sources: JPL Cost Growth Study, D-17868, DOD SPMN, 1999 GSFC-JPL Workshop, JPL staff**
- **Organization: Life cycle activities**
- **Process: Iterative, general principles developed first. Reviewed by managers and developers**
- **Publication: 3rd Quarter FY'00**

Overview of Content

System Definition/System Engineering	12
Planning and Monitoring	18
Cost Estimation	4
Risk Management	3
Organization and Staffing	6
Design and Implementation	12
Integration and Test	16
Configuration Management	3
Software Acquisition	2
Product & Process Verification	4
Flight Software	20
Total	**100**

An Example: Flight Software

- **Margins:**
  - **Goals by development phase (400, 100, 20%)**
  - **Monitoring and validation by measurement**
- **Requirements:**
  - **Accommodate off-nominal inputs from hardware**
  - **AACS algorithm to handle modeling uncertainties and flight events “outside the envelope”**
- **Fault Protection:**
  - **No single-point failure in redundant processing strings**
  - **CDH firmware to incorporate error detection/correction**
  - **Load process to fail if uncorrectable bit error**

An Example: Flight Software (cont.)

- **Test:**
  - Test software to be removable or “rendered harmless”
  - Access to one hardware-in-the-loop testbed
  - All critical testing to be on flight version; tests to be repeated if software is altered

- **Use of Models:**
  - Inter-platform differences minimized and bounded
  - Models used for validation in lieu of tests to be thoroughly characterized

2nd Annual GSFC-JPL Quality Mission Software Workshop

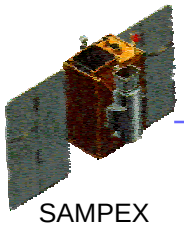
Relationships to Projects

Approaches and Technologies for Flight Software Verification and Validation

Maureen Bartholomew/GSFC

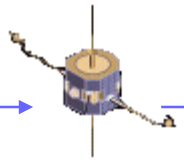
May 17, 2000

FSW Branch Missions



SAMPEX

(launched 8/92)

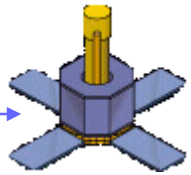


FAST

(launched 6/96)

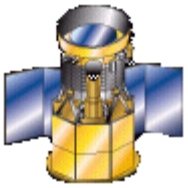
SWAS

(launched 12/98)



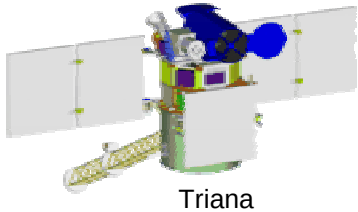
TRACE

(launched 3/98)



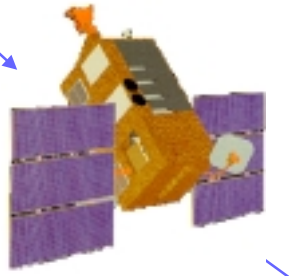
WIRE

(launched 2/99)

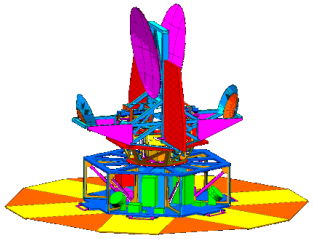


Triana

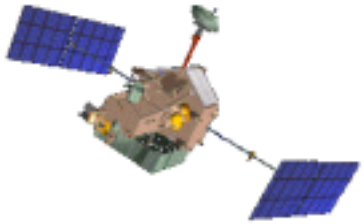
(due to launch 1/01)



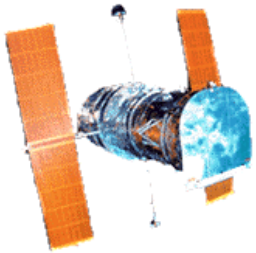
XTE (launched 12/95)



MAP (due to launch 11/00)

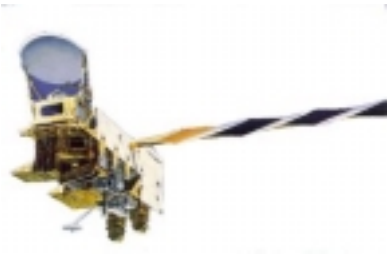


TRMM (launched 11/97)



HST 486 (SM 12/99)

EOS TERRA (launched 12/99)



EOS AQUA (due to launch 12/00)

EOS CHEM (due to launch 12/02)

GSFC FSW Branch Missions

Mission	Characteristics	Test strategy
Sampex, FAST, SWAS, TRACE, WIRE, Triana	IPDT, Single string, **$**	Developers build tested. FSW Maintenance team developed system test scenarios
XTE	IPDT, fully redundant, **$$$**	Dedicated test team (build and system testing). IV&V done by FSW maintenance team.
TRMM	IPDT, fully redundant, **$$$**	Dedicated test team (build and system testing). IV&V done by FSW maintenance team.
MAP	IPDT, Selected redundancy, **$$**	Dedicated test team performing build and system testing

IPDT = Integrated Product Development Team

GSFC FSW Branch Missions

Mission	Characteristics	Test strategy
HST 486	IPDT, Fully redundant, **$$$$**	Dedicated test team performing build and system testing.
EOS Terra, Aqua and Aura	Prime contractor developed, Fully redundant, **$$$$**	GSFC IV&V (test team became FSW maintenance team)

FSW Verification vs. Validation

- Verification: Have we built the system right?
- Validation: Have we built the right system?

- Verification determines whether the software meets system/software specifications
- Validation is concerned with certifying that the system will meet the customer's operational needs

Verification

- Strong FSW verification program requires:
  - Detailed requirements specification
  - Traceability of requirements to FSW tests
  - Early involvement of testers
  - Hi-fidelity testbed (not necessarily a full compliment of hardware/simulators)

Validation

- Strong validation program requires:
  - Well defined operations concepts
    - nominal
    - contingency (including failure recovery)
    - on-orbit maintainability
  - Hi-fidelity testbed(s) with all hardware/simulators
  - Flight software in full flight configuration
  - System and spacecraft subsystem engineering participation

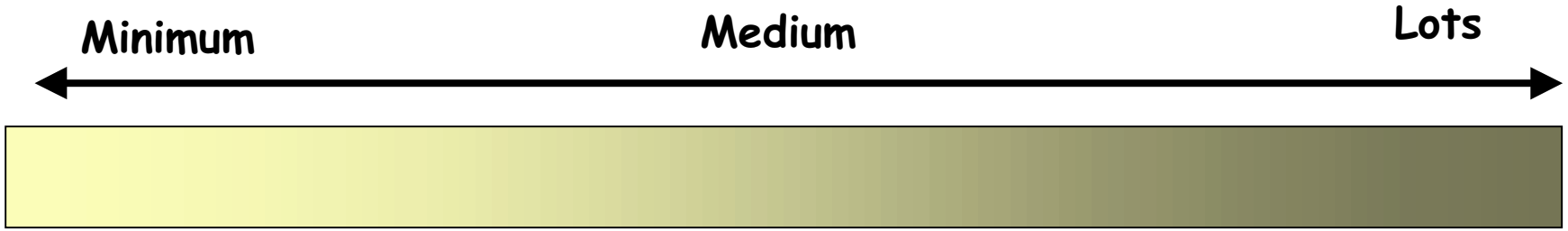
Options for Independence

- **Minimum** - FSW developers test the flight software
  - Build testing done by developers (developer does NOT test FSW functions that he/she wrote)
  - Independent FSW maintenance team leads systems test effort; supported by developers
- **Medium-** dedicated FSW test team
  - Test team is independent from developers
  - Test team responsible for build testing through maintenance
- **Lots -** dedicated FSW test team + independent system test team
  - A dedicated test team responsible for build and system testing
  - Separate test team which independently performs system testing (becomes FSW maintenance team)

Need For Independence?

Apparent cost savings

More test time on the flight software
More "eyes" looking at the system
Early addressing of operations issues
Early addressing of maintenance issues
Early confirmation of hardware
Reduces manpower issues



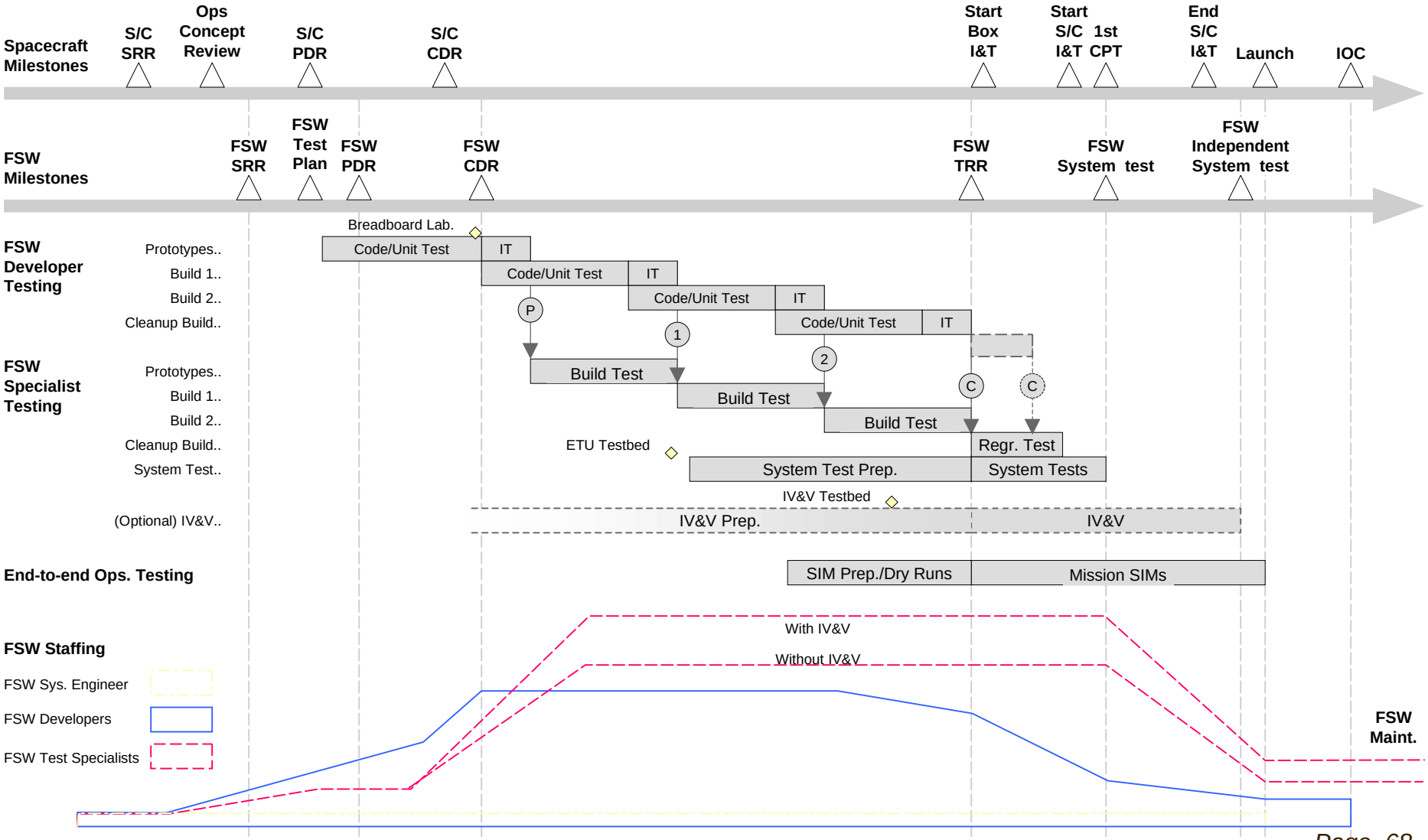
*** Regardless of the "degree" of independence, the key is to have experienced FSW test specialists lead the FSW test effort!!**

Test Levels

	Who	What	Where	Why
Unit Testing	FSW Developers	Logic of individual modules	Breadboards or PCs	Verify the correctness of a FSW "module".
Integration Testing	FSW Developers	Software/ hardware integration	Breadboard testbed	Basic checkout of functionality of a build in preparation for build testing
Build Testing	FSW test Team	Functional and performance requirements	Breadboard testbed	Verify the FSW meets all of the functional and performance requirements
System Testing	FSW test team	Performance/ operational requirements	ETU testbed	Validates FSW, as a whole, can functi in nominal and non-nominal operational conditions.
Spacecraft Testing	Subsystem engineers	Software/ Hardware Interfaces/ Functionality	Spacecraft with flight hardware	Verify flight interfaces and functionality (eg. timing, phasing).
Mission simulations	Flight operators, subsystem enginners, FSW test team	Nominal/ contingency operations.	Operations center/ ETU testbed	Validate flight readiness of the system and operations

Testbed = simulators and/or breadboard/ETU hardware
ETU = engineering test unit(flight-like)

FSW Test Timeline



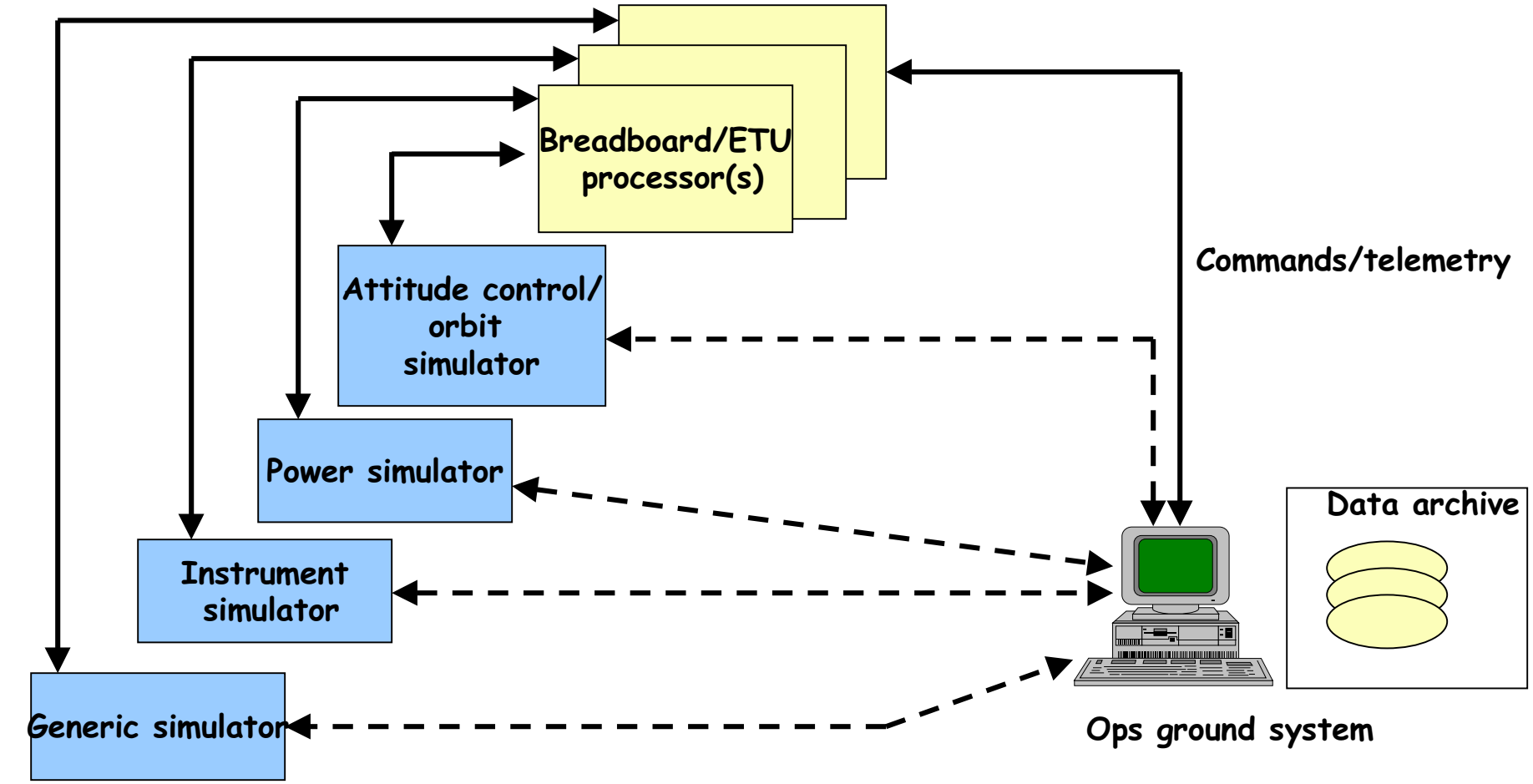
What Makes a Good Test Team?

- Highly EXPERIENCED FSW test lead(s)
- Good mix of people of different disciplines/backgrounds
  - Flight software test specialists
  - Flight software developers
  - Flight Operators
  - On-orbit FSW maintenance specialists
  - Guidance, navigation and control (GN&C) analysts
  - spacecraft subsystem engineers
- Bring the team on early
- Early involvement of systems and spacecraft subsystems engineers

What Makes a Good Testbed?

- Fidelity of the testbeds
  - Fidelity of the processors (timing, processor memory, recorder memory)
  - What is simulated vs. what is “real”
  - Capabilities of the simulators
- Verification of the testbeds
  - Time well spent verifying the testbeds
- Availability of the testbeds
  - Testbeds need to be available and verified BEFORE integration testing begins
  - Need to have enough testbeds to support testing program

Typical Testbed



User Friendly Testbed

- Use of ops ground system during all test phases
- Ability to configure all elements of the testbed from the ops ground system
- All data is telemetered and stored on the ops ground system (including data from simulators/GSE)
- Availability of data analysis/diagnostic tools
- Fast and repeatable test setup

Simulators

- Simulators can make or break test program
  - Not enough emphasis is placed on the requirements and development of simulators
  - Need for simulators to be dedicated to the test program for the life of the mission
  - Need for simulator developers to be available for the life of the test program.
  - Ability to model the spacecraft and hardware environments with hi fidelity
  - Ability to inject faults
  - Flexibility to set any simulation parameters

What is Tested?

- While the focus of FSW V&V is on testing the flight software, out of necessity the following is also tested:
  - Compilers and code generators
  - Hardware (processors, sensors, actuators)
  - Ground system interfaces
    - commands and telemetry
    - page displays
    - table load/dump capabilities
    - stored command loads/dumps

What is Tested? (con't)

- Orbit and attitude products (e.g. Ephemeris, quaternions, alignments)
- Flight software maintenance tools
  - memory load/dump tools and analysis tools
- Simulators
- Ground support equipment (GSE)

Maturity and timely delivery of all of the above elements is critical to the FSW test milestones!

What Makes a Good Test?

- Detailed test scenarios and reviews
- Repeatable test results
  - requires well defined initial conditions. Just one incorrect initial condition can invalidate a test!
- Automated test execution
- Traceable to requirements
- Well documented

Lessons Learned Good Test Processes?

- Reviews
  - detailed scenario
  - test results
- CM
  - FSW version control
  - documentation
  - problem reports
  - tests (procs, logs, data)
- Standards
  - test naming conventions (procs, logs, data)
  - ground database mnemonic names
- Tools

FSW System Testing

- Goal of FSW System testing is to test the operational and performance requirements of the system using an exhaustive set of scenarios
- Two specific types of FSW system tests
  - Failure detection and correction testing
  - stress testing

Failure Detection and Correction Testing

- Consider FDC in all phases of the test program from requirements definition to mission simulations
- Strategize tests for every anomaly
- enable appropriate (I.e. flight-like) detections and corrections during FSW system testing and mission simulations
  - verify no failures in nominal cases
  - verify correct detection and response in failure cases
- Recover from failure condition

Stress Testing

- Goal is to put the system in a realistically stressed configuration
- CPU stressing
  - enable most CPU intensive configuration, considering each task/function
- Throughput stressing
  - intraprocessor communication
    - buffers
    - queues
  - interprocessor communication

Stress Test (con't)

- Throughput stressing (con't)
  - ground/spacecraft
    - maximum commands to the spacecraft
      - memory loads
    - maximum telemetry rate
      - synchronous (science and engineering)
      - asynchronous (i.e. events message, memory dumps)
- Important to consider instrumenting flight software to help facilitate analysis of stress test results

Launch Readiness Prerequisites

1) flight software system testing complete
2) flight software testing at the spacecraft complete
3) operational/ mission simulations complete
4) flight software maintenance team ready

Technologies for Flight Software Verification and Validation

Flight Parameter Database

- Flight systems parameters require constant management
  - achieve consistency between subsystems
  - achieve consistency between flight software, simulators and analysis tools
- Database should contain parameters for:
  - flight software
  - simulators (e.g.. ACS dynamic simulator, power)
  - GN&C algorithm analysis simulator
  - orbit and attitude analysis tools
  - ground
  - other subsystems

Integrated Flight Parameter Database Tool

- WEB based tool allows access (read) by all project personnel
- Assign responsibility and modification rights for individual parameters to the knowledgeable Engineer
- Links between parameters
- Automated generation of FSW (header files), ground database, simulator set-up scripts
- Automatic e-mail notification when parameter changes are made
- History log maintained for all database changes

Automated Tools

- Seamless FSW table load/dump capability (resides in ground system)
  - Define table elements using ground mnemonics
  - table editing and reloading as part of test procedure
  - formatted display of FSW tables
- Plotting tools
  - Scan FSW test data for critical data (e.g. mode transitions, command quaternions)
  - Critical data used to create script file for GN&C algorithm analysis simulator
  - standard set of plots
    - comparison between FSW, attitude control/orbit simulator and the GN&C algorithm analysis simulator
    - Overlay data

Summary

Risk Mitigation

- Use of highly experienced FSW test specialist(s) to lead test program
- Assess level of FSW test independence required for the project
- Use of operations ground system during entire test program
- Define and design failure checks EARLY in FSW development
- Early and active role of FSW maintenance team

Risk Mitigation (Cont'd)

- Early and active role of spacecraft subsystem and system engineers
- Perform mission risk assessment
  - focus on most important aspects of the mission
  - Aids in test progress reporting
- Maximize use of automated tools
- Acquire multiple and quality testbeds

2nd Annual GSFC-JPL Quality Mission Software Workshop

Relationships to Projects
*Summary of Metrics Session**

Mike Stark/GSFC

17 May 2000

***Information presented in this summary will be included in the Workshop Proceedings.**